# Predicting quality of wine based on chemical attributes

Amelia Lemionet, Yi Liu, Zhenxiang Zhou

***Abstract*—Ordinal data structure is difficult to leverage upon due to the fact that it has both the properties of regression and classification. In this paper, we devised a simple new method that leverages upon ordinal data structure of the data: Additive Logistic Regression (ALR). This method is used to predict the quality of Portuguese white wine based on the chemical attributes of the wine. We compare ALR to non-parametric methods. The results showed that our method, albeit simple, out-performs the non-parametric methods.**

## I. Introduction and Related work

ORDINAL regression arises frequently in social sciences and information retrieval where human preferences play a major role [8]. Ordinal responses are comprised of both a ranking structure of real numbers and a discrete structure of classification. This therefore makes the response more difficult to model than real numbers or categorical variables alone. Work has been done recently which makes use of support vector machines to regress on ordinal responses [8] [1], However, we feel that these method have a complex mathematical structure and we believe that a simple and elegant solution can be found.

The data that we decided to test our method on is from the UCI machine learning repository. In this data, the response is the quality of Portuguese white wine determined by wine connoisseurs . There are altogether eleven chemical attributes serving as potential predictors. All predictors are continuous while the response is a categorical variable which takes values from 1 to 10.

In this paper, we propose a new method known as an additive logistic regression and compare its performance to that of weighted linear regression and k-Nearest-Neighbors. The authors acknowledge the fact that work has been done in field of ordinal logisitic regression (order logit)[1] which tackles the problem with similar approaches [2]. However, we like to argue that ordinal logistic regression requires the assumption that the separating hyperplanes between classes must be non-intersecting. While this assumption is theoretically true, it limits the practical application of this method. Additive logistic regression solves this problem and therefore in practice is a simpler method than order logit. In Section II, we will develop the theoretical foundations of the three methods that we will compare in this work: k-nearest neighbors regression, weighted linear regression and additive logistic regression. In Section III, we will analyze the data through exploratory analysis. Afterwards, in Sections IV and V, we implement, adjust, test and compare the three methods at issue. Finally, Section VI and Section VII exhibit the main conclusions and open discussion to other issues of interest.

Advisor: Derek - Professor: Andrew Ng

[1]Ordinal logistic regression finds the probability of the point belonging in each class and assigns the point to the class where the probability is the highest.

## II. Theoretical foundations

Before tackling the wine-quality problem, it is important to introduce the concepts and ideas that we will be using throughout this work.

### A. Notation and Formal statement of the problem

To be consistent, let:

- $m$ be the number of examples in the dataset (this may change as we divide our whole data set into training and testing sub-sets).
- $n = 11$ be the number of features.
- $y$ be the array containing "wine quality" for each of the m examples.
- $X$ be the $m \times n$ matrix containing one example on each row and one feature on each column.

Further on, we will note $\hat{y}^{(i)}$ the prediction we make of $y^{(i)}$.

### B. K-Nearest Neighbor Classification

One method used in ordinal classification in the industry is k-nearest neighbors. An ordinary k-nearest neighbors involves finding the k nearest neighbors of the test data in the variable space and obtain the class for the test data through majority of votes. However, in the case of ordinal classification, in order to make better use of the ordinal structure of the data, we have decided to take the mean of the responses instead of the majority of votes. This is illustrated in the diagram below.

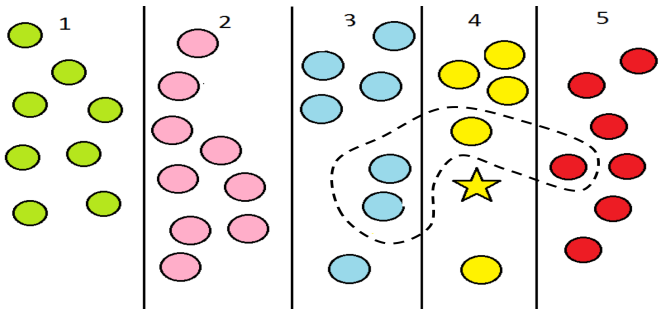


Fig. 1. Diagram Illustration

Consider the star in the diagram as the test point. An the points inside the dotted regions as the k-nearest neighbors (k = 4 in this example). Note that if we consider the majority votes of the nearest neighbors, we would have predicted that this is class 3 instead of class 4. However, if we take the mean, we will have class 3.75 which is much closer to the actual class of the model. There are various ways in which $k$ (the number of neighbors) can be selected. In this work we will use cross validation over the training data and chose $k$ such that the cross validation mean squared error is minimized.

### C. Weighted-Linear Regression

Another method we used in this project is weighted linear regression, which is an non-parametric algorithm. It can often be used to maximize the efficiency of parameter estimation by attempting to give each data point its proper amount of influence over the parameter estimates.

The first weight function we used is:

$$W_{1i} = exp(-\frac{(x^i - x)^2}{2\tau^2}) \tag{1}$$

Where $x$ is the input test point. This weight function captures the distance between the input point and all training points, giving more weights to the training point that is closer to the test point because we assume that similar attributes of wine will have similar quality scores.

The second weight function we used is:

$$W_{2i} = \frac{max(y_i)}{y_i} \tag{2}$$

This weight function considers the magnitude of the response variable, giving more weight to the training response variables with high quality scores.

The last weight function we used is:

$$W_{3i} = \frac{\frac{1}{n}\sum_{i=1}^{n} \text{frequency of } y_i}{\text{frequency of } y_i} \tag{3}$$

This weight function takes into account the nature of the distribution of the response variable. Because the quality scores are not uniformly distributed, there are fewer cases for the scores that are extremely low or extremely high. Hence, this function gives more weight to the training points with low frequency scores.

### D. Additive logistic regression

We generally consider the $y$ to take the form:

$$y = f(x) + \epsilon$$

with $\epsilon$ a zero mean noise. Hence we can predict $y$ as

$$\mathbb{E}[y^{(i)}|X] = f(x). \tag{4}$$

In the case of ordinal responses, we discover a special property of $y$ that we can leverage. To do that, we need to prove the following lemma:

*Lemma 2.1:* Suppose $y$ is a multinomial variable such that $y^{(i)} \in \{0, 1, 2, ..., n\}$

$$\mathbb{E}[y^{(i)}|X] = \sum_{j=1}^{n} \mathbb{P}(y^{(i)} \geq j \mid X) \tag{5}$$

*Proof:*

$$\begin{aligned}\mathbb{E}[y^{(i)}|X] &= \sum_{j=1}^{n} y^{(i)}\mathbb{P}(y^{(i)} = j \mid X) \\ &= \sum_{j=1}^{n}\sum_{k=j}^{n} \mathbb{P}(y^{(i)} = k \mid X) \\ &= \sum_{j=1}^{n} \mathbb{P}(y^{(i)} \geq j \mid X).\end{aligned}$$

■

So essentially, we are turning a multinomial classification problem into a series of binary classification problems. This allows us make use of logistic regression as a method to estimate $\mathbb{P}(y \geq j \mid X)$. Then we propose the following algorithm (Algorithm 1).

**Algorithm 1** Additive Logistic Regression Algorithm

1: **for** j = 0 to n **do**
2: Divide the Training data into $y^{(i)} \geq j$ and $y^{(i)} < j$
3: Estimate $P(y^{(i)} \geq j \mid x^{(i)})$ using logistic regression
4: $\phi_j^{(i)} = \hat{P}(y^{(i)} \geq j \mid x^{(i)})$
5: **end for**
6: $\hat{y}^{(i)} = \sum_j^m \phi_j^{(i)}$

There are two reasons why we choose logistic regression over other methods. First, we note that logistic regression is simple and computationally inexpensive. This is very useful since we are producing n logistic regression models at the same time. If we use more computationally expensive methods such as neural-network, the methods can be too computationally expensive and the time taken for model construction, even with improvement in the results in the end, cannot be justified. Also, using a linear decision boundary prevents overfitting [3] and increases the robustness of our model. This makes logistic regression a better choice compared to the non-linear methods. Second, logistic regression is directly modeling $P(y \mid X)$ and does not make any assumption on the true distribution of the response. As the model between each class is highly correlated, it is hard to justify the use of generative models since we have to justify why the probability distribution remains the same from one class to another. Hence, we have decided to forgo the idea of using generative models.

Although $\hat{y}^{(i)}$ will be a continuous real number (and not a discrete class), this method provides an intuitive classification boundary since it classifies $x^{(i)}$ to be in class $C$ if and only if $\hat{y}^{(i)} \in \left[C - \frac{1}{2}, C + \frac{1}{2}\right]$. [2] The reason that this boundary is used is that we generally classify a point to two different classes at the 0.5 point. However, like binary

[2]Note that this method differs greatly from the ordinal logistic regression in the sense that ordinal logistic regression finds the probability of the point belonging in each class and assigns the point to the class where the probability is the highest. This requires the separating hyperplanes of each of the classes to be perfectly parallel otherwise the method will fail since it might produce negative probabilities (when the probability of the point being lower than $i$ is higher than the probability of being lower than $(i+1)$). Our method however does not rely on the separating hyperplanes being parallel since we are adding the probabilities together and hence the effectiveness of its performance is not affected by the separating hyperplanes.

classification, there might be different cost associated with over-estimating and under-estimating $\hat{y}$. This thought brought us to the possibility of changing the decision boundary of $y^i \in \left[C - \frac{1}{2} - \delta, C + \frac{1}{2} - \delta\right]$ where $\delta \in (-\frac{1}{2}, \frac{1}{2})$.
Before we test the method on a real data set, we decided that we should test this method on a arbitrary data-set with no noise. This helped us understand how the method performs in ideal settings and is a good test of concept. As illustrated in the diagram below, this method has proven to yield desirable results when there is no noise in the data. In a 1-dimensional naive case[3], we can observe that the resulting model fits the data satisfactorily.
Another thing that we have to note is that in the ideal setting, we can observe the number of points in each class is different. The fitting of the graph signals that the method is robust even when having different number of points in each class.

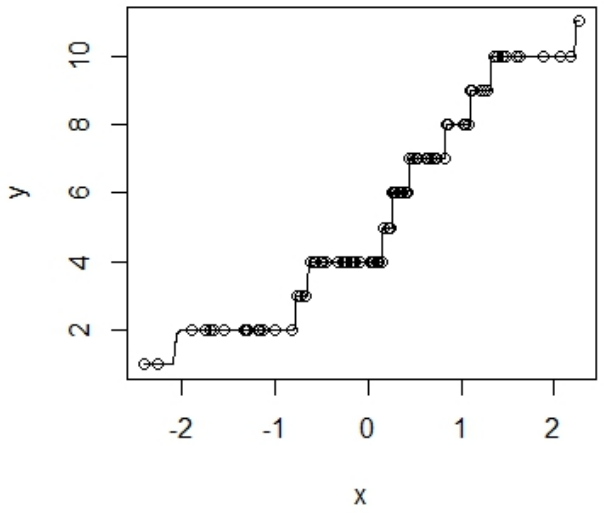


Fig. 2. One dimension illustration

To the authors' limited knowledge, while similar methods exists [1][2][8], the algorithm proposed is not found in literature that we know about.

## III. DATASET AND FEATURES

Our analyses focus in a Portuguese white wine database consisting of $4,898$ observations. The data set contains eleven explanatory variables that measure wine attributes and one response variable: "wine quality". In more detail,

- Fixed acidity: a measurement of the total concentration of titratable acids and free hydrogen ions present in the wine. Theoretically, having a low acidity will result in a flat and boring wine while having too much acid can lead to tartness or even a sour wine. These acids either occur naturally in the grapes or are created through the fermentation process.
- Volatile acidity: a measure of steam distillable acids present in a wine. In theory, our palates are quite sensitive to the presence of volatile acids and for that reason a good wine should keep their concentrations as low as possible.
- Citric acid: one of the many acids that are measured to obtained fixed acidity.
- Residual sugar: measurement of any natural grape sugars that are leftover after fermentation ceases. In theory residual sugar can help wines age well.
- Chlorides: the amount of salt in the wine.
- Free sulfuric dioxide: the free form of SO2 exists in equilibrium between molecular SO2 (as a dissolved gas) and bisulfite ion; it prevents microbial growth and the oxidation of wine.
- Total sulfuric dioxide: amount of free and bound forms of SO2; in low concentrations, SO2 is mostly undetectable in wine, but at free SO2 concentrations over 50 ppm, SO2 becomes evident in the nose and taste of wine.
- Density: measure of density of wine.
- pH: value for pH.
- Sulfates: a wine additive which can contribute to sulfur dioxide gas (S02) levels, which acts as an antimicrobial and antioxidant.
- Alcohol: the percentage of alcohol present in the wine.
- Quality: subjective measurement ranging from 1 to 10 (although the observed data ranges from 3 to 8).

The following table (TABLE I) shows the summary for each variable. Note that all variables are continuous except for the response variable which is categorical.

| Feature | Min | 1st. Q | Median | 3rd. Q | Max | Mean |
|---|---|---|---|---|---|---|
| fixed.acidity | 3.80 | 6.30 | 6.80 | 7.30 | 14.20 | 6.86 |
| volatile.acidity | 0.08 | 0.21 | 0.26 | 0.32 | 1.10 | 0.28 |
| citric.acid | 0.00 | 0.27 | 0.32 | 0.39 | 1.66 | 0.33 |
| residual.sugar | 0.60 | 1.70 | 5.20 | 9.90 | 65.80 | 6.39 |
| chlorides | 0.01 | 0.04 | 0.04 | 0.05 | 0.35 | 0.05 |
| free.sulfur.dioxide | 2.00 | 23.00 | 34.00 | 46.00 | 289.00 | 35.31 |
| total.sulfur.dioxide | 9.00 | 108.00 | 134.00 | 167.00 | 440.00 | 138.40 |
| density | 0.99 | 0.99 | 0.99 | 1.00 | 1.04 | 0.99 |
| pH | 2.72 | 3.09 | 3.18 | 3.28 | 3.82 | 3.19 |
| sulphates | 0.22 | 0.41 | 0.47 | 0.55 | 1.08 | 0.49 |
| alcohol | 8.00 | 9.50 | 10.40 | 11.40 | 14.20 | 10.51 |
| quality | 3.000 | 5.000 | 6.000 | 6.000 | 9.000 | 5.878 |

TABLE I
SUMMARY OF FEATURES AND RESPONSE VARIABLE

The histogram shown in Figure 3, shows that the values for the response variable are not uniformly distributed, and that we have a very high concentration of "average" wines, whereas very low quality and very high quality wines are under-represented.

[3]The splitting point is generated at random

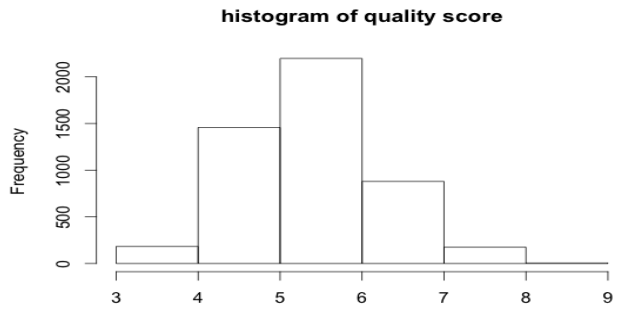


Fig. 3. Distribution of values for "wine quality".

## IV. Fitting the Algorithms

To test the algorithm, we first divided the data into a test set and a training set. We set aside $898$ data-points as the ultimate test set and the rest of the data points $(4,000)$ as the training set. Since $4,000$ data points does not constitute a giant data set, we used cross-validation to find the respective optimal tuning parameters for k-nearest neighbors and weighted linear regression.

We did not normalize the data because most of the data are concentrations measured at the same level and hence the scale is important to the data set.

### A. K-Nearest Neighbors

The first step in adjusting the k-nearest neighbors model was to fix the number of neighbors k. For that, we used 10-fold cross validation and chose k such that the CV residual mean squared error (RMSE) is minimized. This yielded to a result of $k = 9$.

To adapt the mean k-nearest neighbors regression to our ordinal data, we rounded the resulting value to obtain a integer number.

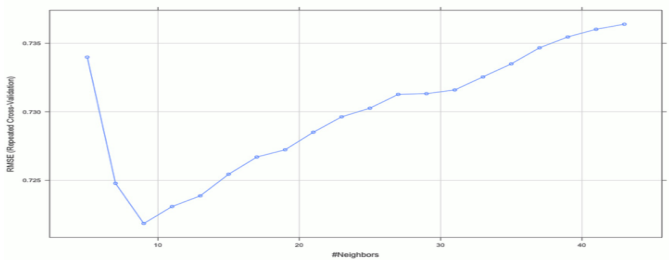


Fig. 4. k = 9 minimizes the cross validation residual mean squared error (RMSE).

### B. Weighted-Linear Regression

In weighted linear regression, since it is a regression algorithm and produces quantitative outputs, we rounded the outputs to the nearest integers to make our final predictions. In weight function (2),(3), it is quite straightforward to fit the model because the weights only depend on the training set and there are no tuning parameters to choose. However, in weight function (1), there are different choices for the tuning parameter $\tau$, which is the bandwidth parameter. This parameter controls how quickly the weight of a training point drops as a function of the distance between $x^i$ and the test point $x$. The larger $\tau$ is, the wider the bandwidth around the test point $x$. To determine the optimal value for $\tau$, we used 10-fold cross validation on the training set and found $\tau = 10$ after taking the minimum CV error.

| Methods | 1-0 LOSS Error Rate |
|---|---|
| Weighted Linear regression with weights (1) | 0.4566147 |
| Weighted Linear regression with weights (2) | 0.4487751 |
| Weighted Linear regression with weights (3) | 0.5124053 |

Fig. 5. Error rate for different weight functions. Normal linear regression for reference.

### C. Additive Logistic Regression

It is important to note that the smallest class is supposed to be 1. In this case, however, the smallest class is 3 which is not ideal. However, a simple solution is to do a parallel shift to the response variable in order to make a prediction and then shift the response back after we make the prediction. This method is mathematically justified since adding a constant does not change the probability.[4]

## V. Results

In classification, one of the key attributes that we consider is the test error rate. In this case, since we are not dealing with binary classification, it is not possible to use the ROC curve or AUC as a criteria to assess our performance. However, in the case of ordinal classification, we can leverage the ordinal nature of the response to consider not only the error rate, but also the percentage of response that is under-estimated and percentage of response that is over-estimated. In this case, the number of times we predicted that the wine had worse quality than it actually has or vice-versa. These are the three key assessment criteria that we will be looking at.

| | ALR | WLR | K-NN |
|---|---|---|---|
| % Test error | 44.65% | 47.55% | 47.33% |
| % overestimate | 26.39% | 31.63% | 32.52% |
| % underestimate | 18.15% | 15.92% | 14.70% |

Fig. 6. Additive Logistic Regression minimizes the test error. The Error represents how many times we miss-classified an observation. The results shown for weighted linear regression (WLR) are the ones corresponding to the weights $W_1$ ($\tau = 10$).

To analyze the previous table, it is important to keep in mind the "naive" scenario. As seen in Fig. 3, the naive predictor would classify all wines as being quality 6 since that is the mode class. If we were to use the naive predictor, the error rate would be of $\sim 55\%$. Compared to the naive scenario, all three methods represent a better alternative.

[4] i.e. $\mathbb{E}(x + a) = \mathbb{E}(x) + a$

Now, when we compare the three methods against each other, additive logistic regression outperforms both k-nearest neighbors and weighted linear regression. Regarding the percentage of over and underestimation, all methods seem to have a tendency to over estimate the response variable.

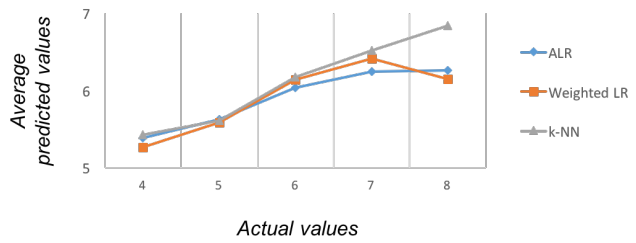


Fig. 7. Predicted values respect order in response variable but most values are concentrated near 6.

## VI. Discussion

In light of all of the methods that we have applied, we are encouraged by the fact that the method that we proposed is performing much better than the other methods. In this Section, we will discuss in detail some of the potential reasons for this. It is surprising that additive logistic regression actually performs much better than k-nearest neighbor classification as k-nearest neighbor can be highly flexible[3]. In this case, we believe that additive logistic regression does better at leveraging the ordinal structure of the data and hence produces better results.

As for weighted linear regression, we need to note that weighted linear regression performs well when the number of predictors is small [3]. In the case of 10 variables, the predictor space may be too sparse to generate good results. (This can also be explained by the curse of dimensionality).

In addition to better accuracy in prediction, we like to introduce an additional benefit using additive logistic regression: its good interpretability. As we go through the ranks, we notice that the importance of each of variables can change. We will illustrate this in the case of our wine data.

We consider the logistic regression model of $P(y \geq 5 \mid X)$

| | Estimate | Std. Error | z value | Pr(>\|z\|) | |
|---|---|---|---|---|---|
| (Intercept) | 106.430556 | 197.457890 | 0.539 | 0.589884 | |
| fixed.acidity | -0.837105 | 0.243918 | -3.432 | 0.000599 | *** |
| volatile.acidity | -2.961208 | 1.665477 | -1.778 | 0.075405 | . |
| citric.acid | 1.894566 | 2.204454 | 0.859 | 0.390105 | |
| residual.sugar | 0.047868 | 0.106767 | 0.448 | 0.653906 | |
| chlorides | -10.198291 | 5.964168 | -1.710 | 0.087279 | . |
| free.sulfur.dioxide | -0.020514 | 0.014467 | -1.418 | 0.156186 | |
| total.sulfur.dioxide | -0.002361 | 0.007583 | -0.311 | 0.755567 | |
| density | -88.031767 | 200.753316 | -0.439 | 0.661019 | |
| pH | -1.800089 | 1.900325 | -0.947 | 0.343510 | |
| sulphates | 3.129435 | 2.527012 | 1.238 | 0.215570 | |
| alcohol | -0.160063 | 0.310396 | -0.516 | 0.606083 | |

Fig. 8. The coefficients for the model $P(y \geq 5 \mid X)$

Notice that the coefficient of fixed acidity in this logistic regression is -0.837105. Fixed acidity in this case is also very important since its absolute z-value is large.

However, consider the logistic regression model of $P(y \geq 7 \mid x)$

| | Estimate | Std. Error | z value | Pr(>\|z\|) | |
|---|---|---|---|---|---|
| (Intercept) | 2.263e+02 | 7.495e+01 | 3.019 | 0.002538 | ** |
| fixed.acidity | 4.621e-02 | 7.721e-02 | 0.598 | 0.549509 | |
| volatile.acidity | -6.270e+00 | 4.568e-01 | -13.726 | < 2e-16 | *** |
| citric.acid | -5.369e-02 | 3.295e-01 | -0.163 | 0.870577 | |
| residual.sugar | 1.510e-01 | 2.864e-02 | 5.274 | 1.34e-07 | *** |
| chlorides | 1.645e+00 | 1.843e+00 | 0.892 | 0.372195 | |
| free.sulfur.dioxide | 1.152e-02 | 3.093e-03 | 3.725 | 0.000195 | *** |
| total.sulfur.dioxide | -2.032e-04 | 1.319e-03 | -0.154 | 0.877623 | |
| density | -2.404e+02 | 7.598e+01 | -3.165 | 0.001553 | ** |
| pH | 1.404e+00 | 3.907e-01 | 3.592 | 0.000328 | *** |
| sulphates | 2.054e+00 | 3.936e-01 | 5.220 | 1.79e-07 | *** |
| alcohol | 7.816e-01 | 9.973e-02 | 7.837 | 4.61e-15 | *** |

Fig. 9. The coefficients for the model $P(y \geq 7 \mid X)$

We notice that the coefficient for fixed acidity here is actually positive and close to zero. This means that the impact of fixed acidity is changing from one class to the other. Due to the special nature of logistic regression, we will be able to observe these changes and therefore peek into the nature of wine tasting through looking at these coefficients. This highly interpretability is not available for the other non-parametric methods that we used in this paper.

In addition, we need to observe that the result separating hyperplane for each class is far from parallel. Although this is counter intuitive, we need to note that that might be other unknown variables (In this case, the year and brand of the wine) which, when working together with our variable, could produce a more or less parallel hyperplane. However, if we use probability of the test point being in a class as the criteria of assigning into that class. (i.e. assign to class with the highest probability) We might face the problem of having negative probability which is not mathematically justified. Additive logistic regression circumscribe this problem and make sure that the predicted results lies within the range of the responses.

However, we would also like to discuss one key part that is lost in additive logistic regression. We have lost the log-odds interpretation of each coefficient, meaning that we cannot translate the coefficients directly to probability.

## VII. Conclusions and Future work

In conclusion, additive logistic regression proves to be more accurate method in terms of prediction accuracy and model interpretability. This is mainly due to the fact that additive logistic regression leverages ordinal nature of the data. However, we still believe that more work can be done in the future. We can add bagging procedure into our methods. Also more work can be done on picking the classification boundary instead of the naive one that we have tried. We believe that such methods would improve the results in terms of prediction accuracy. In addition, we can also treat additive logistic regression as part of general additive models [3]. This might further improves the predictability of our model.

## REFERENCES


[1] Jaime S. Cardoso, Joaquim F. Pinto da Costa. Learning to Classify Ordinal Data: The Data Replication Method Journal of Machine Learning Research 8 (2007) 1393-1429

[2] Harmen J. Dikkers, Support vector machines in ordinal classification, A revision of the abn amro corporate credit rating system August 2005 http://www.kbs.twi.tudelft.nl/docs/MSc/2005/Dikkers_H_J/thesis.pdf.

[3] Trevor Hastie, Robert Tibshirani, Jerome Friedman *Elements of Statistical Learning:Data Mining, Inference, and Prediction.*. Springer, 2nd edition, 2009.

[4] UC Irvine Machine Learning Repository http://archive.ics.uci.edu/ml/.

[5] Winemaker's Academy, understanding wine acidity. http://winemakersacademy.com/understanding-wine-acidity/.

[6] Wine Jargon: What is Residual Sugar?. http://drinks.seriouseats.com/2013/04/wine-jargon-what-is-residual-sugar-riesling-fermentation-steven-grubbs.html.

[7] Red and White Wine Quality https://rpubs.com/Daria/57835.

[8] Wei Chu, S. Sathiya Keerthi, Support Vector Ordinal Regression http://www.gatsby.ucl.ac.uk/~chuwei/paper/svor.pdf

[9] Brian Ripley [aut, cre, cph], William Venables [cph] class: Functions for Classification http://www.stats.ox.ac.uk/pub/MASS4/

[10] Gareth James, Daniela Witten, Trevor Hastie and Rob Tibshirani, ISLR: Data for An Introduction to Statistical Learning with Applications in R, http://www.StatLearning.com

[11] Max Kuhn. Contributions from Jed Wing, Steve Weston, Andre Williams, Chris Keefer, Allan Engelhardt, Tony Cooper, Zachary Mayer, Brenton Kenkel, the R Core Team, Michael Benesty, Reynald Lescarbeau, Andrew Ziem, Luca Scrucca, Yuan Tang, and Can Candan., caret: Classification and Regression Training, https://github.com/topepo/caret/